AFOSR-TR-97-0363

# REPORT DOCUMENTATION PAGE

*Form Approved*
*OMB No. 0704-0188*



| 1. AGENCY USE ONLY *(Leave Blank)* | 2. REPORT DATE | 3. REPORT TYPE AND DATES COVERED |
|---|---|---|
| | 5/23/97 | Final Technical Report 1/15/94- 1/14/97 |

**4. TITLE AND SUBTITLE**

Adaptive Methods for Euler Flows in Complex Geometries

**5. FUNDING NUMBERS**

F49620-94-1-0132

**6. AUTHORS**

Marsha Berger

**7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**

New York University
251 Mercer St.
New York, NY 10012

**8. PERFORMING ORGANIZATION REPORT NUMBER**

FTR1

**9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)**

AFOSR/ NM
110 Duncan Ave., Suite B115
Bolling AFB DC 20332-0001

**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**

**11. SUPPLEMENTARY NOTES**

**12a. DISTRIBUTION / AVAILABILITY STATEMENT**

Distribution Unlimited

DISTRIBUTION STATEMENT A
Approved for public release
Distribution Unlimited

**12b. DISTRIBUTION CODE**

**13. ABSTRACT** *(Maximum 200 words)*

We have developed algorithms for high resolution computations in complicated geometrics that are as automatic as possible. This includes the development of Cartesian grid methods, for easily representing complex geometry using non-body fitted grids, and the development of adaptive techniques, designed specifically for Cartesian grids, for increasing the resolution of a computation where necessary. Some 30 examples are included to illustrate the performance of the method.

19971006 046

accept
11 May 97
27 May

**14. SUBJECT TERMS**

Cartesian Non-body- fitted grid generation

**15. NUMBER OF PAGES**

6

**16. PRICE CODE**

| 17. SECURITY CLASSIFICATION OF REPORT | 18. SECURITY CLASSIFICATION OF THIS PAGE | 19. SECURITY CLASSIFICATION OF ABSTRACT | 20. LIMITATION OF ABSTRACT |
|---|---|---|---|
| UNCLASSIFIED | UNCLASSIFIED | UNCLASSIFIED | UNLIMITED |

# Adaptive Methods for Euler Flows in Complex Geometries

## Final Report REPORT

## AFOSR F 49620-94-1-0132 F49620-94-1-0132

Marsha Berger
Courant Institute
251 Mercer Street
New York, NY 10012

**Abstract**

We have developed algorithms for high resolution computations in complicated geometries that are as automatic as possible. This includes the development of Cartesian grid methods, for easily representing complex geometry using non-body fitted grids, and the development of adaptive techniques, designed specifically for Cartesian grids, for increasing the resolution of a computation where necessary. Some 3D examples are included to illustrate the performance of the method.

# 1 Objectives

We have been developing algorithms to simulate steady state flows in three space dimensions using a Cartesian grid representation of the geometry. This continues to be a collaboration with Captain Michael Aftosmis, at Wright-Patterson/NASA Ames Research Center, and John Melton, at NASA Ames Research Center. In this approach, a solid object is superimposed on an underlying Cartesian grid, and the flow is computed around the object. This makes the problem of volume grid generation substantially easier, with the bulk of the work reduced to finding intersections between a possibly complex configuration and a regular Cartesian grid. However, the difficulty of grid generation is traded for the difficulties in the flow solver of imposing solid wall boundary conditions on a non-body fitted grid. Our work on flow solvers for this kind of grid however indicates that acceptable results that maintain second order accuracy over the entire flow field can be obtained.

Our research in the last several years has focussed on the important issues of robustness, automation, and efficiency, all of which are needed to make this approach usable by a wide audience for a variety of problem geometries. We have developed computational geometry procedures which form the basis of the grid generation algorithms. We have started (but not yet completed) a redesign that allows directional adaptation in the Cartesian mesh generation to improve the efficiency of the overall approach. We have developed numerical discretizations to improve the accuracy of the flow solver at the irregular cut cells adjacent to a solid body.

# 2 Research Accomplishments

## 2.1 Geometry

The ability to automatically generate grids around complex configurations regardless of the topology is extremely important. This permits automatic optimization of inlet design, for example, with no human intervention, even if there are topological changes in the geometry that require new grid generation. Using surface triangulations to describe the geometry, we have developed an approach for generating the volume information needed for a Cartesian grid flow solution. The computational geometry algorithms are well understood, and the grid generation can be fully automated. It consists of basic geometric operations involving planar intersections, rather than case-specific operations that depend on each configuration and are extremely labor-intensive. This approach is more tolerant of skewed surfaces meshes, since they are not used for the flow computations. It permits the easy incorporation of component-based geometry definitions, rather than the single watertight description often required.

To make this approach completely robust, we have developed a procedure to handle the degenerate cases that inevitably arise. For example, the geometry pre-processor needs to determine whether two triangles from different component descriptions intersect. This boils

down to testing the sign of a 4 by 4 determinant. For small determinants, if round-off error leads to an incorrect sign, it is possible to end up with inconsistent answers from neighboring triangles. Ultimately this leads later in the program. There is much recent activity in the computational geometry literature about "exact" geometric computation [9, 5, 7]. Usually this means using extended precision or integer arithmetic, which is very costly. The approach we take, based on the high quality software package by Shewchuk, is to use floating point arithmetic whenever possible, and detect those cases when forward error analysis indicates an untrustworthy. For those cases, the Shewchuk package uses adaptive precision floating calculations to improve the accuracy until the sign computation is robust. If in fact the results of the determinant calculation is exactly zero, this degeneracy is automatically and consistently resolved using the virtual perturbation approach of Mucke.

Based on this approach, a major achievement in this grant period is the development of a geometry pre-processing algorithm which greatly simplifies and speeds up the Cartesian grid generation, as well as being of independent use to other researchers. The preprocessor takes as input multiple component geometries, each a watertight description of an object, but possibly overlapping with other components in an arbitrary way. It produces a single watertight surface description as output. This step also greatly simplifies the grid generator, since later parts of the algorithm will not have to waste time calculating and classifying grid/object intersections that are actually internal. It relieves the CAD operator of a tremendous burden in preparing the geometry for CFD. As an example of this, the geometry in figure 1 was described by 82 components and 320,000 triangles. The component pre-processing took under 4 minutes on an SGI R4000 workstation.

In additional work, we have redesigned the Cartesian mesh generator using advanced data structures and minimal storage. We developed criteria based on estimates of the local curvature to represent a complex geometry to a certain level of accuracy. This is the procedure that determines the level of grid refinement in the initial grid used to start a flow field simulation. (Subsequent refinement is based on properties of the flow field solution rather than the geometry.)

As with any grid generation method, there are difficulties and exceptions which occur rarely, but take most of the programming effort. For the Cartesian grid method the difficulty lies in what we call the "thin section" problem. By this we mean the problem of representing a piece of the geometry that is so thin that it bisects the Cartesian cell into two disjoint sections (see figure 2).

For thin wings for example, this can occur over the last 10-15% of the wing. Without special handling, the upper and lower surface of a good portion of a thin wing or vane could be indistinguishable, giving completely unacceptable results. Viewed as a problem in computational geometry, the solution amounts to recognizing connected components (the intersection of the geometry with a hexahedral cell), and recognizing which side of the geometry is in the flow field (i.e. whether or not you are "inside" or "outside" the geometry).

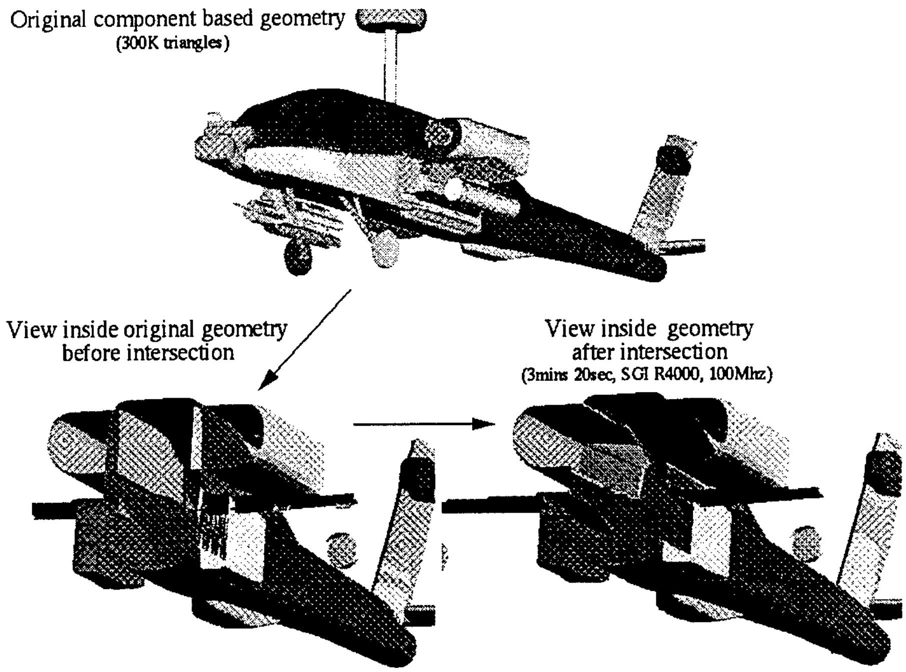


Figure 1: Combat helicopter geometry with 82 components defined by 320,000 triangles.

## 2.2 Flow Solvers

Our Cartesian grid efforts have been directed towards computing steady compressible inviscid flow [6, 2, 1]. We examined the question of accuracy of flow solutions on non-body-fitted grids. Since the cut cells at the boundary of a solid object do not form a smooth grid, how accurately can we compute solutions on such grids? Mesh refinement alone cannot be relied on to improve the accuracy, since the CPU and memory costs of isotropic refinement in 3D are prohibitive. The whole issue of numerical accuracy on irregular grids remains poorly understood. The order of magnitude of the local truncation error can be misleading, since the usual convergence analysis (if you take $n$ steps and make an error $\Delta t h^p$ at each step, the global error is $O(h^p)$) is overly pessimistic [8].

Using a 2D model problem of supersonic flow in a circular channel with an analytic solution we have studied the accuracy of numerical solutions on Cartesian grids. Our results, and those of ([4]), do indeed show that the accuracy of the flow solution suffers at the irregular cells where the body intersects the Cartesian grid. Unfortunately, this is exactly where you would like the solution to be most accurate. For our test problem we used an upwind Godunov scheme with a linear reconstruction of the solution in each cell. The solution over the entire flow field shows second order accuracy, but that drops to 1.3 along the boundaries.

To improve this situation, we developed what might be analogous to p-methods in the

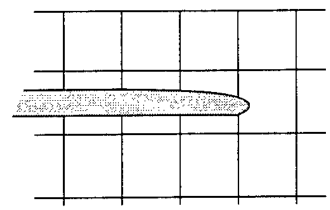

Figure 2: A thin piece of geometry can bisect a cell into disjoint pieces.

finite element world. It is natural to do more work at the boundary, and the asymptotic estimates of the total work over the entire flow field will not be affected (although in practice, a large fraction of the cells are at the boundary). We explored higher order discretizations for use only at the boundaries, including quadratic reconstruction of the solution, two point Guass quadrature for the flux integrations, and a higher order surface boundary condition along with a sub-cell resolution approach were we locally improve the resolution at the boundary. With these improvements, the order of accuracy at the cut cells for our test problem is approximately 1.8, with the magnitude of the error decreased by a factor of 5. (Of course the global accuracy remains second order throughout the flow field). A report on this work is in [3].

## References


[1] M. Aftosmis, J. Melton, and M. Berger. Robust and efficient Cartesian mesh generation for component-based geometry. *AIAA-97-0196*, January 1997.

[2] M. Aftosmis, J. Melton, and M.J. Berger. Adaptation and surface modeling for Cartesian mesh methods. *AIAA-95-1725*, June 1995. 12th AIAA Computational Fluid Dynamics Conference.

[3] M.J. Berger and J. Melton. An accuracy test of a Cartesian grid method for steady flow in complex geometries. In *Proc. Fifth Intl. Conf. Hyperbolic Problems*, 1994. Stonybrook, NY; Also, RIACS Report 95-02.

[4] William J. Coirier and Kenneth G. Powell. An accuracy assessment of Cartesian-mesh approaches for the Euler equations. *AIAA-93-3335*, July 1993.

[5] H. Edelsbrunner and E. Mucke. Simulation of simplicity: a technique to cope with degenerate cases in geometric algorithms. *ACM Symp. on Computational Geometry*, 4, 1986.

[6] J. Melton, M. Aftosmis, M. Berger, and M. Wong. 3D applications of a Cartesian grid Euler method. *AIAA-95-0853*, January 1995.

[7] J. R. Shewchuk. Robust adaptive floating-point geometric predicates. In *Proc. 12th Annual ACM Symp. Comp. Geometry*, May 1996.

[8] B. Wendroff and A. White. Supraconvergent schemes for hyperbolic equations on irregular grids. *Notes on Numerical Fluid Mechanics*, 24, 1989.

[9] C. Yap. Symbolic treatment of geometric degeneracies. *Journal of Symbolic Computation*, 10, 1990.